**Technical Design Document**

**Name:** Matthew Pocrnic  
**Date Created:** 6/15/2025

## **Program Description:**

This program allows the user to enter a list of their monthly expenses. The user enters the type of expense and the amount for each item. When they are finished, the program uses the reduce() method to calculate the total expense, find the highest expense, and the lowest expense. It then displays a summary showing these results with labels.

## **Functions used in the Program:**

### **1. Function Name: get\_expenses**

**Description:** Collects the user's expense entries, each consisting of a description (type of expense) and its amount, until the user indicates completion.

**Parameters:** None

**Variables:**

* expenses (list) – List of tuples containing (expense\_type, amount)
* expense\_type (str) – The type/category of the expense
* amount (float) – The numeric cost of the expense

**Logical Steps:**

1. Display instructions for entering expenses.
2. Use a loop to repeatedly ask the user for the type of expense and its amount.
3. Append each (expense\_type, amount) as a tuple to the expenses list.
4. Allow the user to type 'done' to end input.
5. Return the completed list of expenses.

**Returns:**

* expenses (list) – List of tuples (expense\_type, amount)

### **2. Function Name: main**

**Description:** Controls the execution of the program, calling get\_expenses() and performing calculations using the reduce() function. Prints a summary of results.

**Parameters:** None

**Variables:**

* expenses (list) – List of expense tuples from get\_expenses()
* total (float) – Total of all expense amounts
* highest (tuple) – Tuple of the highest expense: (expense\_type, amount)
* lowest (tuple) – Tuple of the lowest expense: (expense\_type, amount)

**Logical Steps:**

1. Call get\_expenses() to retrieve the list of expenses.
2. Use reduce() to compute the total expense amount.
3. Use reduce() to determine the highest expense (by amount).
4. Use reduce() to determine the lowest expense (by amount).
5. Display the total, highest, and lowest expenses with labels.
6. Handle the case when no expenses are entered.

**Returns:** None

### **Logical Steps (Program Flow):**

1. Start the program by calling main().
2. Prompt the user to enter expense names and amounts repeatedly.
3. Collect all entered expenses in a list.
4. Calculate the total expense using reduce().
5. Calculate the highest expense using reduce().
6. Calculate the lowest expense using reduce().
7. Print the summary results for the user.
8. End the program.

**Link to my COP2373 repository:** [**here**](https://github.com/mpocrnic/COP2373)

Screenshot of output from running code

